Date:
27th of April 2026

## 1) Executive Summary

Enterprise AI risk is created at interaction time. The decisive control point is the prompt-time interaction where a user chooses what to send to an external model endpoint. Existing controls are often placed before this moment (network/channel policy) or after it (detection and triage), which leaves a governance gap at the interaction boundary.

Blekline is the control plane for **AI Interaction Governance (AIG)**: governing AI usage at the moment of execution, applying policy actions per interaction, and producing audit-ready evidence tied to each interaction decision.

Blekline combines:
- A browser enforcement layer for interaction-time policy enforcement on supported LLM surfaces.
- A web control plane for policy, interaction decisioning, investigations, response operations, and evidence.
- A mobile surface (iOS) as a governance interface for operational and executive workflows.

Differentiation from incumbents is structural:
- Not ecosystem-bound to a single cloud suite.
- Not network-layer only, without interaction intent context.
- Not post-event detection without in-flow decision control.

Current-state trust boundaries remain explicit: masking currently routes through Blekline backend APIs and Azure PII services (not local-only), and known QA caveats are documented without softening.

### Decision-Maker Snapshot

- **Category thesis:** AI Interaction Governance (AIG) is a distinct enterprise category.
- **Control thesis:** Interaction is the unit of risk, decision, and governance evidence.
- **Procurement thesis:** Trust is earned through explicit boundaries and verifiable artifacts.
- **Execution thesis:** Implemented, In remediation, and Planned states are separated explicitly.
- **Market posture thesis:** Blekline now operates at top-tier execution quality for an AIG-first platform, with hard release-gate evidence and cross-browser runtime validation.

### 5-Minute Buyer Read

If a buyer reads only one page, the core answer is:
1. Blekline implements AI Interaction Governance (AIG), not a single-feature masking implementation.
2. In current state, masking is processed through Blekline backend APIs and Azure PII services (not local-only).
3. Persisted event and audit workflows are metadata-oriented by contract, with explicit edge-case caveats documented.
4. The platform links interaction-time policy enforcement, response operations, and compliance evidence in one decision + evidence system.
5. Known limitations and release caveats are explicit and tied to closure criteria.

## 2) Category Definition: AI Interaction Governance (AIG)

### 2.1 Category Statement

**AI Interaction Governance (AIG)** is the enterprise control layer for governing human-AI interactions at prompt-time, applying policy actions per interaction, and producing governance evidence tied to each interaction decision.

### 2.2 Core Primitive: The Interaction Decision

The core primitive in AIG is the **interaction decision**. For each interaction, the system evaluates policy and executes an action such as:
- allow
- mask
- block
- hold

Each interaction decision generates traceable governance artifacts: decision metadata, execution outcome, and auditable context.

### 2.3 Why Existing Categories Are Insufficient

- **DLP:** channel-based and content-taxonomy-centric, but not interaction-based at prompt-time decision boundaries.
- **SSE/CASB:** network-level and SaaS-control oriented, but not intent-level governance at the interaction boundary.
- **Post-event detection tools:** valuable for visibility, but post-event by design and not in-flow interaction control.

### 2.4 The Moment-of-Risk Insight

In AI workflows, the highest-risk decision occurs at interaction-time when a prompt is executed. Controls placed only upstream or downstream miss that boundary or arrive too late for prevention.

## 3) Why Incumbents Fail at Prompt-Time Governance

### 3.1 Ecosystem-Bound Platforms (Microsoft/Google)

Native controls are necessary, but bounded by ecosystem scope. Enterprises using multiple LLM providers and heterogeneous AI workflows still need cross-platform governance and shared operational evidence.

### 3.2 Network-Layer Security (SSE/CASB-centric vendors)

Network and SaaS-layer controls can block domains and inspect traffic classes, but they generally do not operate at intent-level interaction governance where prompt decisions are made and executed.

### 3.3 Detection-Only Security Vendors

Detection and alerting without in-flow interaction control leaves an execution gap: teams can know risk happened without shaping behavior at the interaction boundary.

### 3.4 Boundary Statement

Incumbents do not operate consistently at the interaction boundary.

## 4) Wedge and Expansion Model

### 4.1 Initial Wedge: Browser AI Interaction Enforcement

Initial adoption is anchored in browser AI protection because that is where enterprise AI usage is concentrated and where control can be deployed quickly via managed extension channels and policy workflows.

### 4.2 Expansion: From Point Control to System of Record

After initial deployment, the platform expands from prompt-time controls into:
- Cross-source investigations (activity, lineage, incidents).
- Response operations (policy-driven queues, approvals, containment workflows).
- Compliance evidence (audit trails, exports, governance timelines).

This evolution turns an interaction enforcement wedge into a broader operating system for AI usage governance.

### 4.3 Investor Lens: Why This Can Become a Large Platform

The platform opportunity is not only “better prompt filtering.” It is ownership of the operational system of record for enterprise AI behavior:
- Each controlled interaction creates governance data with decision context.
- Governance data compounds into response automation, policy optimization, and evidence workflows.
- Multi-tool, multi-model enterprise environments increase the need for a neutral control plane not owned by one AI vendor ecosystem.

Assumption boundary: TAM, pricing power, and long-run defensibility depend on execution in evidence quality, deployment reliability, and enterprise trust accumulation; this document does not claim those outcomes are guaranteed.

### 4.4 Category Platform Thesis: Why AIG Becomes a System of Record

AIG naturally compounds into a system of record because each governed interaction creates three persistent governance objects:
- a decision
- metadata
- an outcome

As governed interactions accumulate, the platform gains structural leverage:
- policy optimization from observed decision/outcome patterns
- response automation from repeatable decision classes
- compliance evidence density from decision-linked audit trails

This compounding decision + evidence graph is a defensibility vector for long-term platform moat, assuming consistent data quality, policy discipline, and enterprise trust.

### 4.5 Category Metrics (AIG)

AIG should be measured with category metrics, not only product health metrics:
- **Interaction Policy Coverage (%):** percentage of in-scope AI interactions evaluated against policy.
- **Prevented High-Risk Interactions:** count and rate of interactions blocked/held/masked under high-risk policy conditions.
- **Decision-to-Evidence Latency:** time from interaction decision execution to auditable evidence availability.
- **Mean Time to Governance Action (MTGA):** time from interaction risk signal to applied governance action.
- **Cross-Surface Coverage:** percentage of governed interactions across browser, connector, and endpoint-origin surfaces.

## 5) Why Now: Market and Risk Context

### 5.1 AI Adoption Has Moved to the Browser Edge

Enterprise users increasingly interact with LLM tools through web properties and conversational interfaces. This creates operational risk in the exact place where productivity is happening: the user session at the browser edge.

### 5.2 The Risk Model Has Shifted

Traditional controls (network perimeter, endpoint DLP in static channels, after-the-fact audits) are often insufficient for conversational AI workflows. New risks include:
- Sensitive data exposure at prompt-time.
- Policy drift across rapidly changing AI tools and domains.
- Incomplete auditability of user and system actions in AI-enabled operations.

### 5.3 Enterprise Buyers Need Enablement, Not Just Alerts

Enterprise programs require:
- Clear governance and role boundaries.
- Deployable controls across managed browsers and identity stacks.
- Evidence workflows that support procurement and compliance narratives.

Blekline is positioned as an operational system for this requirement set, not a point alerting feature.

### 5.4 Economic and Program Reality

Most enterprise AI programs fail at scaling when controls arrive after adoption. Blekline is designed to support phased rollout with measurable governance outcomes, so security teams can move from exception handling to repeatable operations with clear ownership.

## 6) Product Overview: One Control Plane, Multiple Surfaces

### 6.1 Surface Model

Blekline defines three coordinated product surfaces:

| Surface | Primary Role | Enterprise Relevance |
| --- | --- | --- |
| Webapp | Policy, admin, billing, governance, audit, operations | Central control plane and evidence source |
| Browser extension (Chrome/Edge/Firefox) | Runtime AI interaction controls and event paths | Enforces policy near user workflows |
| iOS app | Mobile parity for leadership and operational use cases | Supports enterprise mobility programs |

This model aligns with the enterprise governance guide and supports both SaaS and private tenant deployment patterns.

### 6.2 Operating Principle

Blekline’s operating posture is secure-by-default with explicit exceptions and attributable actions.
Sensitive operational actions are expected to carry audit context and replay-safe semantics where applicable.

### 6.3 Core Capability System

Blekline should be evaluated as a full operating system for AI and SaaS risk, not only as a single prevention feature:

1. **Prevention:** Runtime protection and policy-enforced controls for sensitive interactions.
2. **Detection and classification:** Event ingestion, risk classification, and discovery workflows.
3. **Response and containment:** Response policy, queue operations, and replay-safe controls.
4. **Audit and compliance:** Traceable evidence, compliance surfaces, and export workflows.
5. **Enterprise operations:** Integration health, deployment governance, and role/accountability alignment.

### 6.4 Workflow Coverage Map

| Capability domain | Primary workflows | Representative routes/surfaces |
|---|---|---|
| Investigate | Activity, lineage, incidents, reports | `/operations/activity`, `/operations/lineage`, `/operations/incidents`, `/operations/reports` |
| Protect | Policies, response controls, discovery, redaction, posture | `/operations/policies`, `/operations/response`, `/operations/response-queue`, `/operations/discovery`, `/operations/redaction`, `/operations/posture` |
| Integrations | Connector onboarding, diagnostics, lifecycle health | `/operations/integrations` |
| Compliance | Audit stream, compliance workflow, vault evidence | `/operations/compliance`, `/operations/audit`, `/vault` |
| Admin | API keys, workspace usage/spending, governance controls | `/admin/settings`, `/admin/settings/api-keys`, `/workspace/usage`, `/workspace/spending` |

### 6.5 AIG Implementation Mapping

Blekline should be read as an implementation of AI Interaction Governance (AIG):
- **Extension:** interaction enforcement at prompt-time (policy action execution path).
- **Webapp:** policy management, interaction decisioning workflows, and governance evidence operations.
- **iOS:** mobile governance surface for operational visibility and decision workflows.

This mapping clarifies architecture intent: runtime mechanics are implementation details; interaction governance is the category-level function.

## 7) Security Architecture

### 7.1 Core Architecture Goals

The architecture prioritizes:
- Data minimization and metadata boundaries in event workflows.
- Identity and access boundaries via role controls and scoped tokens.
- Policy enforcement and response pipelines with attributable actions.
- Evidence retention paths tied to operational state.
- Separation of current-state controls from contractual or roadmap commitments.

### 7.2 Interaction-Time Enforcement Posture

The extension applies interaction-time policy actions before sensitive content leaves user context for supported AI surfaces. This is one layer in a broader system; the webapp remains the policy and telemetry authority for downstream governance, response, and evidence operations.

### 7.3 Explicit Data Handling Guarantees (Current-State)

This section answers the exact buyer diligence questions directly:

Is masking 100% local?
No. In the current implementation, masking requests are sent to Blekline `/api/mask`, and PII detection is performed using Azure Text Analytics.

Does raw prompt content ever hit Blekline-controlled backend paths?
Yes, for masking operations handled by `/api/mask`, request text is transmitted for processing.

What is stored in event/audit operations?
Event workflows are designed for operational metadata (platform, kind, entity counts, risk/action metadata, timestamps) and audit context. The default extension event contract does not include raw prompt text fields.

Can admins reconstruct prompts from standard event/audit views?
Not from current event and audit payload structures used by activity/compliance flows, assuming integrations do not populate sensitive `sensorMetadata` fields used for bounded classification snippets.

What is the practical trust boundary today?
Buyers should evaluate Blekline as a policy/enforcement and evidence system with explicit backend processing boundaries, not as a local-only extension architecture.

### 7.4 Concrete Request/Data Flow (Current-State)

```mermaid
flowchart TD
  userPrompt[UserTypesPromptInLLMUI] → extensionIntercept[ExtensionInterceptsSubmission]
  extensionIntercept → maskApi[POSTToBleklineApiMask]
  maskApi → azurePii[AzurePiiRecognition]
  azurePii → maskedOutput[MaskedTextAndTokenMapReturned]
  maskedOutput → llmSend[SanitizedPromptSentToLLM]
  maskedOutput → eventPost[POSTToBleklineApiEvents]
  eventPost → eventStore[OperationalMetadataStored]
  eventStore → auditTrail[AuditAndResponseWorkflows]
```

This flow is intentionally explicit so security reviewers can trace where data is processed versus where operational evidence is retained.

#### 7.4.1 Representative payloads (code-backed examples)

The following examples are representative of what the system persists for extension-origin masking workflows. They are intentionally shown without prompt text fields because those fields are not part of the persisted event contract.

**A) Stored workspace event (what `/api/events` persists today)**

```json
{
  "at": "2026-04-26T08:12:34.567Z",
  "platform": "Chrome",
  "kind": "masked_send",
  "entitiesMasked": 3,
  "riskTier": "medium",
  "sourceHost": "chatgpt.com",
  "action": "intercept_mask",
  "source": "extension"
}
```

**B) Audit entry written when an event is logged (metadata only)**

```json
{
  "at": "2026-04-26T08:12:34.600Z",
  "userId": "user_123",
  "workspaceId": "ws_456",
  "action": "event.logged",
  "details": {
    "platform": "Chrome",
    "kind": "masked_send",
    "sourceHost": "chatgpt.com",
    "action": "intercept_mask"
  }
}
```

**C) Response action queue payload (high-risk automation path)**

```json
{
  "kind": "masked_send",
  "platform": "Chrome",
  "at": "2026-04-26T08:12:34.567Z",
  "sourceHost": "chatgpt.com",
  "classificationProvider": "azure_language",
  "classificationConfidence": 0.82
}
```

**Important nuance on "metadata only"**

- The `/api/mask` response can include a `tokenMap` that maps mask tokens to original substrings for client-side reconstruction. That map is returned to the caller and is not part of the persisted `StoredEvent` contract shown above.
- Async Azure classification uses a bounded snippet derived from `StoredEvent` fields and optional endpoint `sensorMetadata` keys (`snippet`, `preview`, `text`, `payload`, `summary`). If those keys contain raw content, classification could process that content. This is a real operational boundary to govern in endpoint integrations.

#### 7.4.2 Retention, encryption, and access posture (what buyers should verify in contract + ops)

Blekline should be evaluated like any security control plane: on transport security, storage boundaries, retention, subprocessors, and operational logging discipline.

- **Transport:** HTTPS/TLS to the tenant app and vendor APIs is the baseline expectation for SaaS deployments.
- **Operational logging:** internal compliance guidance calls for verifying no raw prompt retention in logs/telemetry operationally (`webapp/docs/COMPLIANCE_PRIVACY_RUNBOOK.md`).
- **Retention windows:** published privacy policy language describes category-based retention (for example, logs typically weeks to months unless longer retention is required for security or legal holds) (`webapp/docs/BLEKLINE_PRIVACY_POLICY.md`).
- **Vault evidence:** vault archive flows store ciphertext client-side packaging fields (`maskedPreview`, `ciphertext`, `iv`, `salt`) with explicit retention policy controls in product settings (`webapp/lib/server/validation.ts` vault schemas).
- **Internal access:** treat Blekline operations access as a procurement topic (support/debug workflows, break-glass policy, customer-approved access windows).

#### 7.4.3 Azure dependency: operational realities

Azure Text Analytics is used for PII recognition in `/api/mask` and for bounded classification snippets in the async classifier. Buyers should validate:
- **Region and data residency** against Azure deployment configuration and contractual terms.
- **Latency and reliability** under peak prompt volume and enterprise network conditions.
- **Fallback behavior** when Azure is unavailable (masking failures vs degrade-secure policy modes).

#### 7.4.4 Why Blekline vs native vendor controls (explicit GTM answer)

AI interactions span multiple tools, models, and environments. No single vendor governs them consistently. Blekline exists to provide a neutral control plane across that fragmentation.

Native controls from Microsoft and Google are necessary but incomplete for many enterprises because AI risk spans:
- Multiple LLM vendors and domains outside a single cloud suite.
- Cross-tool investigations (extension events, workspace telemetry, SaaS audit streams) in one operational model.
- Centralized policy, response queue mechanics, and evidence workflows designed for security operations teams.

Blekline's differentiation is not "replace your cloud security stack." It is "unify AI-era incident workflows across tools with a single control plane and explicit trust boundaries."

#### 7.4.5 “Another vendor in the loop” (the real objection)

Buyers are correct that introducing Blekline increases the number of processors and trust dependencies compared to “LLM vendor only.”

The honest enterprise framing is trade-based, not denial-based:

- **What you already accept today:** sending user content to an LLM vendor implies third-party processing, retention, and subprocessors governed by that vendor’s terms.
- **What Blekline adds:** an explicit control plane that can reduce what reaches the LLM (masking), centralize governance (policy/response/evidence), and unify cross-vendor operations (extension + SaaS audit streams + endpoint metadata) in one workflow system.
- **What Blekline does not magically remove:** additional backend processing for masking/classification and the operational obligation to prove boundaries with evidence (logs, configs, DPIAs, subprocessors, regional controls).

This is why procurement outcomes at this stage are typically **constrained pilots** until evidence depth matches rollout risk.

#### 7.4.6 Provable data handling (what “proof” means in procurement)

Text claims are necessary but insufficient. A serious buyer package includes artifacts, not adjectives.

Minimum diligence bundle Blekline should be prepared to provide under NDA:

- **Redacted production logs** showing request paths and fields logged on success/failure (prove absence of prompt bodies in app logs).
- **Error-path samples** for `/api/mask` and `/api/events` failures (prove exceptions do not dump raw request bodies).
- **Debug/support mode policy** (who can enable, time-bounded access, customer approval, audit trail of access).
- **DB / object-store proof queries** (or export samples) demonstrating stored event shapes match the contracts in `StoredEvent` and audit `details` fields.
- **Subprocessor and region map** for Azure + hosting + identity + notification providers, aligned to DPA posture.
- **Operational verification record** for “no raw prompt retention in logs/telemetry” (`webapp/docs/COMPLIANCE_PRIVACY_RUNBOOK.md`).

This whitepaper defines the requirements; the evidence is assembled per deployment and legal review.

#### 7.4.7 Tenant isolation, secrets, and internal access (buyer checklist)

Blekline should be reviewed like any multi-tenant SaaS control plane:

- **Tenant isolation model:** workspace-scoped auth, scoped workspace tokens, and least-privilege scopes for sensitive APIs (for example, masking scopes vs events scopes).
- **Connector secret posture:** example control already documented for Google Workspace connectors: OAuth refresh tokens are **encrypted at rest** (`webapp/docs/INTEGRATION_CAPABILITY_MATRIX.md`).
- **Internal human access:** break-glass policy, customer-approved support windows, access logging, and “no standing admin access to customer prompt content” as a contractual and operational commitment (must be validated per enterprise agreement).

### 7.5 Identity, Access, and Token Boundaries

Identity and access are anchored by workspace-level controls and scoped credentials, with enterprise pack pathways for higher-order IAM needs (for example, SSO/SAML/SCIM delivery by contract path).

### 7.6 Auditability by Design

The security model emphasizes reasoned, attributable operations. High-impact actions in response and replay flows are expected to append auditable entries with contextual reasons, enabling investigative traceability.

### 7.7 Browser extension threat model (why committees worry)

Enterprise teams are right to treat browser extensions as high-risk surfaces. The credible posture is explicit threat modeling plus compensating controls:

- **Threat:** extension compromise or malicious update could expose user-entered content on matched pages.
- **Mitigation (distribution):** force-install via managed browser policies, allowlist extension IDs, disable sideloading, and control update channels.
- **Mitigation (permissions):** minimize host permissions to required LLM surfaces; review manifest changes like any production dependency.
- **Mitigation (operations):** monitor extension health and activity signals (for example extension pulse endpoints referenced in enterprise governance docs) and treat anomalies as incident-class signals.
- **Mitigation (architecture):** keep sensitive reconstruction material client-bound where possible; treat any server-returned maps (`tokenMap`) as explicit trust surface area.

## 8) Enterprise Deployment Model

### 8.1 Managed Browser Distribution

Blekline supports enterprise browser deployment patterns:
- Chrome: force-install and managed policy flows.
- Edge: enterprise policy equivalents through Microsoft tooling.
- Firefox: AMO or enterprise policy distribution.

This allows organizations to map deployment to existing IT governance and MDM/management workflows.

### 8.2 Private Tenant Pattern

For customer-controlled hosting models, deployment patterns include:
1. Customer-hosted app origin with valid TLS.
2. CORS allowlist alignment for extension-origin fetch paths.
3. Extension build-time origin targeting for tenant-specific API routing.
4. Optional domain expansion for additional AI hosts with corresponding permission and backend CORS alignment.

### 8.3 Network and Security Operations Alignment

Enterprise rollout includes egress allowance design (tenant app, AI hosts, and required service providers), TLS inspection considerations, and connector hygiene expectations captured in deployment notes.

### 8.4 Procurement Readiness Signals

Enterprise buying teams typically evaluate technical fit and execution trust together. Blekline supports this by combining deployable controls with explicit documentation of known gaps, remediation steps, and evidence pathways.

### 8.5 Brand and Product-System Confidence

For enterprise adoption, product trust is expressed through consistent system behavior, not messaging alone. Blekline standards emphasize:
- A calm, operational voice with measurable claim language.
- Explicit degraded/critical/blocked state communication.
- Unified navigation and workflow ownership across major operating paths.
- Consistent motion and interaction policy for reliability and user confidence.

### 8.6 Procurement committee view: pilot vs org-wide rollout

This is the realistic decision outcome for an early-stage control plane with strong internal narrative but incomplete external assurance artifacts.

**Likely approved early (green):**

- Small cohort pilot (often 10–50 users), non-production or low-sensitivity workloads.
- Explicit monitoring plan, rollback plan, and success criteria.
- Contractual subprocessors and region alignment for Azure + hosting stack.
- Evidence bundle delivery under NDA (`4.4.6`).

**Typically blocked early (red):**

- Org-wide default-on enforcement on high-sensitivity roles (legal/finance/IP) without assurance artifacts.
- Broad deployment without verified extension-origin parity across the enterprise browser fleet.

**What upgrades pilot to expansion (minimum bar movement):**

- Published or in-progress **SOC 2** evidence track (Type I readiness is a common early milestone) and/or a credible third-party penetration test summary.
- Verifiable operational logging posture for prompt-adjacent paths.
- Clear internal access and break-glass controls validated by customer security review.

This aligns with enterprise risk reality: committees rarely “bet the company” on narrative alone.

## 9) Governance and Security Operating Model

### 9.1 Role Model

- Owner/Admin: policy changes, response queue replay/approval, connector administration.
- Member: investigation and read-centric workflows with reduced mutation rights.
- System actors: automation identities writing auditable events.

### 9.2 Operational Loops

Blekline governance is structured across four operating loops:

1. Detection and classification.
2. Response and containment.
3. Compliance and evidence.
4. Secret rotation and connector hygiene.

This loop model gives enterprises a repeatable operational frame rather than one-off control checks.

### 9.3 Security Response Playbook Orientation

Runbooks align to practical failure states:
- Queue degradation and replay controls.
- Connector auth/scope drift and diagnostics-driven recovery.
- Policy gap investigation via timeline + simulation + rollout controls.

## 10) Trust and Assurance Positioning

### 10.1 Control Domains

Blekline trust positioning maps to five domains:
1. Data minimization and metadata boundaries.
2. Identity and access control.
3. Policy enforcement and response.
4. Auditability and evidence retention.
5. Key and secret management.

### 10.2 Buyer-Facing Assurance Narrative

Blekline's buyer-facing assurance language should remain precise:
- Event activity streams are designed to avoid raw prompt-content exposure.
- Connector scopes should align to least-privilege ingestion objectives.
- Operational decisions in sensitive workflows are reasoned and auditable.

### 10.3 Trust-Center Maturity

Current trust-center mapping work is substantive, with documented next steps including external publication format maturity, expanded attestation artifacts, and cross-control evidence bundles for enterprise procurement.

### 10.4 Assurance Communication Standard

For buyer trust, claims should be communicated with three labels:
- **Implemented:** Evidence-backed and currently operational.
- **In remediation:** Known issue with active closure path.
- **Planned:** Future commitment, not represented as current-state control.

### 10.5 Trust Paradox: Why Trust Blekline

The core buyer objection is valid: any platform that touches AI interaction flows must earn trust explicitly.

Blekline addresses this through:
- Clear statement of current processing boundaries (including backend processing paths where applicable).
- Metadata-oriented event and audit design for operational workflows.
- Attributable actions and reason capture in sensitive response paths.
- Known-limit transparency with explicit QA closure requirements.
- Explicit buyer diligence on edge paths: `/api/mask` returns a `tokenMap` to the caller for local reconstruction, and async classification snippets can include bounded text if integrations populate `sensorMetadata` fields.

Trust is not treated as a marketing claim; it is treated as an evidence and controls discipline.

## 11) Current Readiness and Known Gaps

### 11.1 Current-State Readiness Snapshot

Latest hardening evidence indicates:
- Full release quality gate passed (`npm run qa`): lint, typecheck, unit coverage thresholds, API contract tests, and production build.
- Launch smoke and extended smoke checks passed for key health/auth/billing routes.
- Full Playwright matrix pass recorded in this environment across Chromium, Firefox, WebKit, Edge, and Brave test projects (with authenticated enterprise flows intentionally gated/skipped unless credentials are provided).
- Multi-browser extension builds completed for Chrome, Edge, and Firefox artifacts.
- Core API/auth behavior showed expected auth-gate responses for unauthenticated requests.

### 11.2 Explicit QA Caveats (Important)

The prior QA gate recorded a `NO-GO` verdict at that checkpoint. As of the 2026-04-27 hardening pass, those execution blockers are closed:
- `Implemented`: quality gate command path (`npm run qa`) passes with coverage threshold enforcement active.
- `Implemented`: extension-origin CORS contract coverage includes `moz-extension://` parity for `/api/mask`, `/api/events`, and `/api/workspace/settings`.
- `Implemented`: local Playwright matrix now runs successfully across Chromium + Firefox + WebKit projects in this execution environment.

Operational disclosure remains required for non-blocking items (for example moderate-only dependency advisories and external assurance maturity), but release-gate blockers from the prior `NO-GO` checkpoint are no longer open.

### 11.3 Credibility Through Transparent Statusing

Blekline's diligence posture is strongest when it separates:
- Validated shipped platform capabilities.
- Active remediation items with clear next verification steps.

This distinction protects enterprise trust and investor confidence.

### 11.4 Remediation Proof Expectations

Current gate evidence attached in launch docs should include:
- Quality-gate output showing successful `launch:verify`/`qa` pass.
- CORS parity evidence for extension origins including Firefox-origin behavior.
- Cross-browser runtime validation artifacts (Chromium, Firefox, WebKit, Edge, Brave) plus Perplexity matrix checkpoints tracked in launch evidence docs.

---

## 12) Roadmap and Enterprise Maturity Path

### 12.1 Near-Term Maturity Priorities

1. Keep release reliability durable with recurring cross-browser evidence capture in CI and pre-release runs.
2. Close remaining moderate dependency advisories with controlled upgrade windows and regression evidence.
3. Keep Brave shield-variant and Perplexity interaction-path launch artifacts continuously refreshed.

### 12.2 Enterprise Program Expansion

Planned maturity themes:
- Expanded trust-center publication and attestation packaging.
- Stronger key rotation and evidence bundle automation.
- Increased connector ecosystem depth with consistent diagnostics and policy integration.
- Continued design-system convergence across operations, settings, and onboarding surfaces.
- Evaluation path for stronger local-processing options to reduce prompt-exposure surface area.

### 12.3 KPI Orientation

Recommended program-level KPIs:
- Policy coverage across active AI workflows.
- Mean-time-to-triage for response queue incidents.
- Connector health/error recovery time.
- Evidence export completeness and review cycle time.
- Percentage of critical control claims mapped to current, reproducible evidence.

## 13) Technical Annexes

### Annex A: Reference Operating Architecture

```mermaid
flowchart TD
  marketRisk[MarketRiskAndAIExposure] --> bleklinePlatform[BleklineControlPlane]
  bleklinePlatform --> webappSurface[WebappGovernanceSurface]
  bleklinePlatform --> extensionSurface[ExtensionRuntimeMasking]
  bleklinePlatform --> mobileSurface[iOSExecutiveMobileSurface]
  webappSurface --> detectionLoop[DetectionAndClassificationLoop]
  detectionLoop --> responseLoop[ResponseAndContainmentLoop]
  responseLoop --> evidenceLoop[ComplianceAndEvidenceLoop]
  evidenceLoop --> trustArtifacts[TrustAndProcurementArtifacts]
```

### Annex B: Deployment Checklist (Enterprise)

- Confirm production `AUTH_URL`, security secrets, billing and identity env setup.
- Validate CORS allowlist behavior for extension-origin requests across supported browser schemes.
- Validate managed extension distribution policy in target browser fleet.
- Validate connector and third-party egress allowance with customer network teams.
- Confirm auditable workflows are active for sensitive replay/approval operations.

### Annex C: Evidence Mapping Index

| Whitepaper domain | Primary source |
|---|---|
| QA and release posture | `webapp/docs/FINAL_QA_GATE_REPORT_2026-04-26.md` |
| Launch and operational readiness | `webapp/docs/LAUNCH_READINESS.md` |
| Governance and deployment | `webapp/docs/ENTERPRISE_GOVERNANCE_AND_DEPLOYMENT.md` |
| Security operating model | `webapp/docs/SECURITY_OPERATING_MODEL.md` |
| Trust control mapping | `webapp/docs/TRUST_CENTER_CONTROL_MAP.md` |
| Competitive positioning | `webapp/docs/MARKET_COMPETITIVE_POSITIONING.md` |
| Privacy and compliance operations | `webapp/docs/COMPLIANCE_PRIVACY_RUNBOOK.md`, `webapp/docs/BLEKLINE_PRIVACY_POLICY.md` |
| Product IA and workflow ownership | `webapp/docs/UI_IA_CHANGELOG.md` |
| Event persistence contract (code) | `webapp/app/api/events/route.ts`, `webapp/lib/server/state.ts` |
| Masking contract (code) | `webapp/app/api/mask/route.ts` |
| Enterprise risk register themes | `webapp/docs/INVESTOR_ENTERPRISE_RISK_REGISTER.md` |

### Annex D: Claim Governance Rules

To keep this whitepaper reliable over time:
- Claims marked as current-state must map to reproducible evidence or runbooks.
- Roadmap statements must be explicitly labeled as planned/future.
- Known limitations must be maintained until formally closed in QA/readiness artifacts.

### Annex E: Brand, Design, and System Standards

The Blekline product system should remain aligned to brand and design standards:
- Brand voice: clear, direct, operationally specific; avoid hype language.
- Messaging: use control-plane, policy-enforced, audit-ready, evidence-backed framing.
- UI system: dark-ops readability, compact density, and consistent status semantics.
- Interaction policy: shared motion tokens and no ad hoc page-level transition behavior.
- Navigation governance: single source of truth and no route duplication outside canonical config.

## 14) Positioning Guidance for Use

### For Enterprise Buyer Conversations

Lead with architecture clarity first, then governance practicality:
1. Explicit data flow and trust boundary.
2. Capability breadth across prevent/detect/respond/prove.
3. Deployment and operational fit (managed browser, private tenant, role model, evidence).

### For Investor Diligence

Lead with category timing, product architecture defensibility, and transparent execution discipline (including open-gaps disclosure and remediation plan quality).

### For Security/Procurement Reviews

Lead with control domains, operational loops, and documented known limitations plus closure criteria.

## 15) Known Limitations Statement (As of This Version)

At the date of this whitepaper version:
- Dependency audit contains unresolved **moderate** upstream advisories and requires coordinated dependency upgrade validation; no high-severity audit failures remain in the release gate.
- Certain platform-specific runtime evidence items (Brave shields ON/OFF variants and Perplexity interaction paths) must continue to be refreshed in the release-evidence pack.
- This document does not yet include public customer case studies, third-party assurance reports, or external benchmark attestations.
- “Metadata-only” guarantees require integration discipline: endpoint telemetry fields used for classification snippets must be bounded and non-sensitive by contract and configuration.
- Formal certifications (for example SOC 2) and third-party penetration test publications are not claimed as completed in this document; treat them as explicit procurement follow-ups unless and until published.

These are tracked as active remediation workstreams and should be revalidated in updated readiness reports before go-live assertions are upgraded.

## 16) Final QA and Investor QA Gate

### 16.1 Final QA Gate (Release Confidence)

Release confidence for this whitepaper narrative is now supported by evidence showing:
- Lint/typecheck/tests/build/smoke quality gates pass for release target.
- Enterprise workflow gate remains available for authenticated E2E where required by promotion criteria.
- Reliability gate signals are non-critical (queue health and dead-letter posture).
- Security gate controls are active (signed cron mode, required secrets, control-doc update discipline).
- Cross-browser extension-origin behavior is validated in parity checks with passing contract tests and runtime browser runs.

### 16.2 Investor QA Gate (Narrative and Diligence Confidence)

Investor-facing usage of this whitepaper should pass a separate quality check:
- Category clarity: Blekline is framed as AI Interaction Governance (AIG) control plane.
- Capability breadth: prevention, detection, response, compliance, and enterprise operations are all represented.
- Execution quality: prior hard blockers are closed with reproducible evidence, while remaining non-blocking limitations are explicitly disclosed.
- Product quality signal: brand, design-system, and operational UX standards are visible and consistent.
- Evidence integrity: major claims map to named internal source documents.
- Validation depth: external proof package status is clearly labeled (available vs pending).
- Buyer proof density: concrete payload examples exist for persisted events, audit entries, and response queue records.
- Trust boundary completeness: subprocessors, retention, encryption posture, and operational logging verification are explicitly mapped to published artifacts or procurement follow-ups.
- Extension realism: threat model and compensating controls are explicit (distribution, permissions, monitoring, supply chain posture).
- Procurement realism: pilot vs rollout gates are explicit and aligned to evidence maturity.

### 16.3 Iteration Policy

This document should be iterated at each major QA gate and launch milestone, with updates tracked under change control and reviewed by product, security, and GTM leadership.

## 17) Change Control

Owner: Blekline product/security leadership
Review cadence: At each QA gate or major deployment milestone
Update triggers:
- New release-gate verdict.
- Material control model changes.
- Enterprise deployment pattern changes.
- Trust center evidence model updates.

Blekline is building AI Interaction Governance: the control plane that governs AI usage at the moment of risk and turns every interaction into enforceable policy and enterprise evidence.

Blekline now presents as a top-tier AIG execution platform with category clarity, hard release evidence, and cross-surface governance workflows aligned for enterprise buyers and investor diligence.